Algorithms for Real-Time Processing

A. Farina
Technical Directorate, Radar & Technology Division
Alenia Marconi Systems
Via Tiburtina km. 12.400, 00131 Rome, Italy

Tel: +39-6-41502279, Fax: +39-6-41502665, e-mail: afarina@amsjv.it

Key words: parallel processing, QR decomposition, Inverse QR decomposition, lattice flow graphs, CORDIC.

1. SUMMARY AND INTRODUCTION

This paper describes methodologies for on-line processing of received radar data by a set of N antennas and M pulse repetition intervals (PRIs) for the calculation of space-time adaptive (STAP) filter output. The numerically robust and computationally efficient QR-decomposition (QRD) is used to derive the so called MVDR (Minimum Variance Distortionless Response) and lattice algorithms; the novel inverse QRD (IQRD) is also applied to the MVDR problem. These algorithms are represented as systolic computational flow graphs. The MVDR is able to produce more than one adapted beams focused along different angular directions and Doppler frequencies in the radar surveillance volume. The lattice algorithm offers a computational saving; infact, its computational burden is $O(N^2M)$ in lieu of $O(N^2M^2)$. An analysis of the numerical robustness of the STAP computational schemes is presented when the CORDIC (CO-ordinate Rotation DIgital Computer) algorithm is used to compute the QRD and the IQRD. Benchmarks on general purpose parallel computers and on a VLSI (Very Large Scale Integrated) CORDIC board are presented.

The paper is organized as follows. Section 2 introduces the STAP problem and the basic systolic algorithms to reach real time adaptation. The lattice and its generalization (vectorial lattice) for wideband problems are discussed in section 3. The IQRD based MVDR algorithm is presented in Section 4. Experiments of mapping the basic triangular systolic array onto general purpose parallel computers are discussed in Section 5. The use of VLSI CORDIC board is the theme of Section 6. The MVDR systolic algorithm is applied to off-line process recorded live data; the detection of vehicular traffic is shown (Section 7). Finally, Section 8 gives a perspective for future research.

2. BASELINE SYSTOLIC ALGORITHM

The detection of low flying aircrafts and/or surface moving targets, and the stand-off surveillance of areas of interest require a radar on an elevated platform like an aircraft. The AEW (Airborne Early Warning) radars pose a number of interesting technical problems especially in the signal processing area. The issue is not new: detect target echoes in an environment crowded of natural (clutter), intentional (jammer), and other unintentional RF (especially in the low region of microwaves, e.g. VHF/UHF bands) interference. The challenge is related to the large dynamic range of the received signals, the non-homogeneous and non-stationary nature of the interference, and the need to fulfil the surveillance and detection functions in real time. One technique proposed today to solve the problem is based on STAP [2], [7], [8], [14] to [16] . Essentially, the radar is required to have an array (for instance, a linear array along the aircraft axis) of N antennas each receiving M echoes from a transmitted train of M coherent pulses. Under the hypothesis of disturbance having a Gaussian probability density function and a Swerling target model, the optimum processor is provided by the linear combination of the NM echoes with weights $\mathbf{w}=\mathbf{M}^{-1}\mathbf{s}^*$, envelope detection and comparison with threshold. $\mathbf{M}$ is the space-time covariance matrix, i.e. $\mathbf{M}=E\{\mathbf{z}^*\mathbf{z}^T\}$ where $\mathbf{z}$ (dimension NMx1) is the collection of the NM disturbance echoes in a range cell, $\mathbf{s}$ - the space-time steering vector - is the collection of the NM

Paper presented at the RTO SET Lecture Series on "Military Application of Space-Time Adaptive Processing", held in Istanbul, Turkey, 16-17 September 2002; Wachtberg, Germany, 19-20 September 2002; Moscow, Russia, 23-24 September 2002, and published in RTO-EN-027.

samples expected by the target, and (*) stands for complex conjugate. A direct implementation, (via Sample Matrix Inversion, SMI) of the weight equation $\mathbf{w}=\mathbf{M}^{-1}\mathbf{s}^*$ is not recommended. One reason is related to the poor numerical stability in the inversion of the interference covariance matrix especially when large dynamic range signal is expected during the operation; another one is the very high computational cost. There is a need of extremely high arithmetic precision during digital calculation. Note that double precision costs four times as much as single precision. The situation would be different if, instead of operating on the covariance matrix **M**, we would operate directly on the data snapshots $\mathbf{z}(k)$, k=1,2,...,n, where n is the number of snapshots (i.e. : range cells) used to estimate the weights **w**. It can be shown that the required number of bits to calculate the weights, within a certain accuracy, by inversion of **M** is two times the number of bits to calculate the weights operating directly on the data snapshots $\mathbf{z}(k)$. This is so because the calculation of power values is avoided thus the required dynamic is halved. The algorithms that operate directly on the data are referred to as "data domain algorithms" in contrast to the "power domain algorithms" requiring the estimation of **M**.

The QRD is a numerical technique to solve least-square problems, like the one in STAP, that avoids direct computation and inversion of interference covariance matrix [1], [3]. Indicate with **Z** the n x (NM)-dimensional matrix which collects the n data snapshots:

$$\mathbf{Z}(n) = [\, \mathbf{z}(1)\, \mathbf{z}(2) \ldots\ldots\ \mathbf{z}(n)]^T \tag{1}$$

The weight equation can be written as follows:

$$\mathbf{Z}^H(n)\, \mathbf{Z}(n)\, \mathbf{w} = \mathbf{s}^* \tag{2}$$

where $(\bullet)^H$ stands for complex conjugate transpose. Taking the data matrix **Z** and operating on it with unitary (i.e. covariance preserving) matrix **Q** (with dimension nxn) we are able to transform the matrix **Z** in an upper triangular matrix **R** (with dimension NMxNM):

$$\mathbf{Q}\,\mathbf{Z} = \begin{bmatrix} \mathbf{R} \\ \mathbf{0} \end{bmatrix} \tag{3}$$

thus equation (2) can be rewritten as:

$$\mathbf{R}^H\, \mathbf{R}\, \mathbf{w} = \mathbf{s}^* \tag{4}$$

which is now easily solved by forward and back-substitution steps as follows. Indicating by a new vector **t** the product **R w**, equation (4) becomes:

$$\mathbf{R}^H \cdot \mathbf{t} = \mathbf{s}^* \tag{5}$$

this can be solved in **t**. Subsequently, the additional equation:

$$\mathbf{R}\, \mathbf{w} = \mathbf{t} \tag{6}$$

is solved in **w**. A noticeable improvement of the basic technique allows to calculate the STAP output without extracting the weights, i.e: without performing the two substitutions above (see, for instance [1] at page 147). In summary, either the weight vector **w** or the output signal of the STAP are obtained without forming and inverting any covariance matrix. By using a large number of bits the data domain algorithm provides the same results of the power domain algorithm which estimates the covariance matrix $\mathbf{Z}^H(n)\, \mathbf{Z}(n)$ and derives the weight vector by the conventional Cholesky factorisation of that matrix in equation (2). The noticeable result is related to the far superior performance of the data domain algorithm when using a limited number of bits; in

fact, data domain needs half number of the bits required by the power domain method to reach good interference cancellation and target coherent integration.

The triangularization of the data matrix, see eqn. (3), can be done with the following known methods: Givens rotations, Householder reflections (a generalization of Givens rotations) and Gram Schmidt [1]. Another method to obtain a sparse (actually a diagonal in lieu of triangular) data matrix is the singular value decomposition; the Jacobi and Hestenes are recursive parallel algorithms to efficiently obtain the SVD. The Lanczos is another numerically efficient candidate to solve our real-time STAP problem [4]. The preferred approach in this paper is the one based on Givens rotations. All these methods enjoy the possibility to be mapped onto a parallel processor like a systolic array. This means that the algorithm is readily transformed in a computer architecture; this is not the case for the equation (2) where a single processor computer has the task to perform the indicated operations. Today it is possible to implement the systolic array with custom VLSI technology thus providing compact processors requiring limited prime power. An additional advantage is related to the large data throughput of the parallel processor representing a suitable mean to reach the real-time operation.

The baseline architecture considered for the STAP problem is the trapezoidal one depicted in Figure 1 [3]. This constitutes the generalization of a method, which was originally developed for MVDR beam forming, by QRD. The NM-dimensional triangular array ABC receives the snapshots of data from a set of range cells and outputs from the right-hand side the matrix **R** produced as the data descend through the array. The matrix is passed to the right-hand column of cells DE which serves to steer the main beam in the desired angular direction and Doppler frequency. Multiple beams can be formed simply by adding right-hand columns as depicted in Fig. 1; they are constraint post-processors. The bulk of the computation, i.e. the QRD, is common to all of the separate beam forming tasks, and only needs to be performed once.

The MVDR processor in Fig. 1 is designed to operate in the following manner [3]. The triangular processor, in normal adaptive mode (selected by setting a - not shown - input binary flag f=1), is fed with sufficient data snapshots to form a good statistical estimate of the environment. The triangular array is then frozen (by setting the input binary flag f=0) while a look-direction constraint is input as though it were a data vector $\mathbf{z}(n)$ emerging from the multi channel tapped delay line. This serves to calculate the vector $\mathbf{a}= (\mathbf{R}^H)^{-1}\ \mathbf{s}^*$ which is captured and stored in the right hand column (also operating in mode f=0); this vector needs to determine the STAP output $\mathbf{e}(n)=\mathbf{z}^T(n)\mathbf{R}^{-1}(\mathbf{R}^H)^{-1}\ \mathbf{s}^*$. Once the vertical columns have been initialized, the adaptive mode of operation (f=1) is selected for both the main triangular array and the right hand columns and more data snapshots are presented to the processor. The processor then updates its estimate of the environment (via the stored quantities **R** and **a**) and simultaneously outputs the STAP signals from the bottom of the column DE.

The number of processing elements in the triangular systolic arrays is $0.5\ M^2N^2$. The MVDR algorithm has a noticeable computational advantage with respeet to the SMI which requires $O(N^3M^3)$ arithmetic operations per sample time. Two types of processing elements are needed within the triangular array: one calculates the sine and cosine of an angle between two input data values, the other rotates the remaining data of the same angle. The calculation of the rotation and the application of the rotation is repeated for each row of the triangular array. A third cell type is used in the look-direction constraint columns. Every processing cell of the triangular array should perform on average 24 floating point operations per data snapshot. Let d be the desired data rate, i.e. the snapshots per second to process, the systolic machine should perform $12\ dM^2N^2$ flops. As an example, let d be 1 MHz and NM=100 the corresponding processing power needed is 100 Gflops approximately. By down sampling the radar data of a factor ten, the required processing power is 10 Gflops.

3. LATTICE AND VECTORIAL LATTICE ALGORITHMS

An advanced processing architecture referred to as MVDR lattice processor requires $O(N^2M)$ arithmetic operations per sample time ; it is described in [3]. It takes advantage of the time-shift invariance associated with STAP. The data entering the triangular array changes very little from one PRI to the next which means

that a large part of the computation is being repeated on successive PRIs albeit in different parts of the array. This repetition is eliminated in the lattice algorithm where the big trapezoidal array is decomposed in a lattice of smaller (i.e.: of dimension N) trapezoidal arrays; the lattice has M stages (see figure 2). The lattice based MVDR operates in a similar manner to the big trapezoidal array; details are found in [3]. If M=N=10 and the update rate is one tenth of 1 MHz, the required computational power is 1 Gflop. The lattice algorithm has also been designed and tested with simulated data for wideband STAP [10]; this architecture is particularly useful: (i) to deal with wideband radar, (ii) to compensate for amplitude and phase mismatching between the receiving channels, and (iii) to combat the hot clutter. The processing architecture, named vectorial lattice, operates on an array of N antennas, M PRIs and P samples taken within the radar range cell. The lattice has again M stages, each having trapezoidal arrays of dimension NP. The computational complexity of the scheme is $O(MN^2P^2)$.

In the above mentioned processing architectures, namely: the MVDR, the lattice and the vectorial lattice, the common processing module is the triangular systolic array. In the Sections 5 and 6 to follow we report the results concerning the mapping of the triangular systolic array onto parallel processors.

4. INVERSE QRD BASED ALGORITHMS

A further improvement of the triangular systolic array for STAP processing is called IQRD (Inverse QR Decomposition) and promises an additional decrease of the required computational power.

The need to reduce the computational requirements of the triangular systolic array was discussed in Section 2. It was mentioned the possibility to down sample of a factor ten, say, the update of the triangular array. This tacitly requires the extraction of the adaptive weights of the STAP at the low update rate and the application of the weights to the radar echoes at the natural rate of the data. This approach has the following practical problem. The MVDR systolic array of Figure 1 could extract the adapted weights via back substitution; however, pipelining the two steps of triangular update and back-substitution seems impossible. There are two possibilities to overcome this problem. The first is to use a triangular array in addition to the main one; the second triangular array being reversed with respect to the array which updates the matrix **R**. This approach requires more hardware to be integrated on the chip. The second approach exploits a recursive equation which updates $(\mathbf{R}^H)^{-1}$ instead of **R**. The update of $(\mathbf{R}^H)^{-1}$ serves to the purpose of extracting the weights. This algorithm, referred to as IQRD, can be implemented with just one triangular systolic array, which has a specular orientation of the basic triangular array to update **R** (see figure 3). A limitation of this approach is related to the difficult schedule of the various processing steps. A detailed comparative analysis of the IQRD and QRD based MVDR algorithms is presented in [12]. Also an implementation of the corresponding systolic architectures with the use of the CORDIC algorithm, as basic building block is discussed.

5. EXPERIMENTS WITH GENERAL PURPOSE PARALLEL PROCESSORS

This section summarizes the findings described in details in [6]; today this study seems out of date for the advancement in the signal processing hardware, nevertheless it is still very instructive. We study the use of parallel processors of MIMD (Multiple Instruction streams Multiple Data streams) and SIMD (Single Instruction stream Multiple Data streams) types available on the market (early 1990s). This approach is meant to be propaedeutic to the VLSI solution. In fact, it provides guidelines for the design of the processing architecture to be implemented on silicon. The problems of synchronization of the whole systolic array by a master clock and the data transfer between processors can also be investigated. Additionally, an estimate of the computational performance of several candidate processing architectures is also possible.

With reference to the MIMD machine, a re-configurable Transputer based architecture (the MEIKO Computing Surface, using up to 128 T800 INMOS Transputers) has been adopted and three solutions have been proposed. The first uses a ring of Transputers. Then an improvement of performance is reached by

diminishing the amount of communication required; such a result has been achieved by using a linear array of processors. The mapping of the algorithm on a triangular array of processors has also been studied. This solution allows the use of an arbitrary number B of processors provided that $B = p(p+1)/2$, p being an integer number. This mapping shows performance better than the linear mapping. The investigation on MIMD computers is concluded with a comparison of the results achieved by using the nCUBE2 with 64 processing elements.

With reference to SIMD machine, tests on the Connection Machine CM-200 and the MasPar MP-1 have been performed. CM-200 is equipped with 8192 single bit processors, whereas MP-1 has 4096 four bit processors. The QRD has been mapped onto a mesh architecture for both machines.

Without going into the details, which are described in [6], the main conclusions of the work are the following. The experimental work done suggests to map the dependence graph of the systolic array for the QRD algorithm on a MIMD machine configured as a triangular array. An achievable data throughput is in the order of 10 KHz for a STAP with MN=16 and 120 PEs using the MEIKO Computing Surface. A data throughput in the order of 100 KHz should be feasible either with advanced Transputers or with devices like the Texas TMS320C40. These conclusions, which date back eight years ago, should be reconsidered in the light of the more powerful COTS (Commercial Off The Shelf) machine available today. Table 1 summarizes pros and cons of the hardware implementation of STAP based on COTS.

6. EXPERIMENTS WITH VLSI BASED CORDIC BOARD

To explore the possibility of achieving better computational performance and using compacter systems - for installation in an operational radar - a QRD algorithm has been mapped on an application specific prototyping platform which contains four VLSI CORDIC ASICs and some FPGAs (Field Programmable Gate Arrays) [9]; this work was done in co-operation with the Technical University of Delft (The Netherlands).

The test-bed platform mainly consists of a large (modular) memory buffer that is connected to a Sun Workstation via a VME bus. The memory buffer stores data that flow through the application board, back into the buffer. The application board consists of four CORDIC processors which are mesh-connected. These four processors perform complex rotations on two-dimensional complex vectors. The CORDIC processor is a pipeline processor operating in block floating point. The physical connections between the CORDIC has been implemented via Xilinx chips. In the benchmark described in [9], the triangular systolic array was mapped on the 2x2 CORDIC application board of the tested platform. This four CORDIC mesh corresponds functionally to one of the processing nodes constituting the triangular systolic array. However, as the CORDIC processors are pipelined processors, many of these rotations can be performed at a very high throughput rate (the clock rate of the pipe), provided a schedule can be found such that the pipe can be kept filled. Such a schedule can indeed be found for the QRD algorithm.

The results of the benchmark may be briefly summarized as follows. With a 100% pipeline utilization of the CORDIC, the throughput can be computed simply as,

$$\text{Throughput} = \frac{\text{clockfreqCORDIC}}{\text{number of rotations}} \tag{7}$$

where "clockfreqCORDIC" is the clock frequency of the CORDIC processor (only 5 MHz in the experiments, just to show that no extremal values are needed), and "number of rotations" is the number of rotations (vectorizations included) for the case where we simulate a system of MN degrees of freedom.

For MN=10 the throughput is approximately 80 KHz, i.e. 80000 input vectors could be processed per second, which is better than the results reported in Section 5 where larger computers and higher clock frequencies were used. In a non-experimental implementation of the CORDIC system described in this Section, clock

frequencies up to 40 MHz are easily achievable; this would improve the throughput even further within a factor of 8.

Table 2 summarizes the pros and cons of the hardware implementation of STAP based on custom VLSI. Selection between COTS and VLSI is still an open question; the specialized technical literature reports descriptions of experimental systems using both the two technologies: a consensus has not been found yet on which technology to use, even though the trend seems today in favor of COTS.

7. PROCESSING OF RECORDED LIVE DATA

The data recorded by the Naval Research Laboratory (NRL-USA) airborne multi-channel radar system have been processed by a systolic trapezoidal array which implements the STAP [7]. The performance of the algorithm have been evaluated against ground clutter, littoral clutter and jammer. Vehicular traffic has also been detected. The systolic array processing has been emulated with a MATLAB software tool.

The airborne radar system used by NRL for its STAP flight test program is a modified AN/APS-125 surveillance radar; the operating frequency is 420-450 MHz. The array consists of 10 hooked dipole antennas spaced approximately a half wave length apart, mounted in a 90 degree corner reflector to provide elevation pattern shaping. The two outer dipoles are terminated yielding eight channels with roughly equivalent element patterns and 3 dB beam widths of 80 degrees for both azimuth and elevation. The array was energized with a high power corporate feed which applied a taper on transmit such that the maximum azimuth sidelobe level is 25 dB down with respect the main beam.

The receiving system consists of 8 identical channels with each channel having a UHF preamplifier, mixer, VHF amplifier band pass filter and a synchronous demodulator. The synchronous demodulator consists of two demodulators, one referenced to the coherent oscillator (COHO) and the other referenced to the COHO shifted by 90 degrees. This yields two bipolar video channels, one in phase (I), the other quadrature phase (Q). Each I & Q signal is converted to digital by a 10 bit, 5 MHz analogue to digital converter. The radar Pulse Repetition Frequency (PRF) is 300/750 pps.

The output of the receiving system is 16 digital channels for a total digital word width of 160 bits with a clock rate of 200 nsec. This yields a data bandwidth of 800 Mbps which is buffered in real-time and stored in magnetic tape.

7.1 SYSTOLIC ALGORITHM FOR LIVE DATA PROCESSING

As indicated in Figure 1, the radar has an array of N=8 antennas and receiving channels. Each of these receives M echoes from a transmitted train of M (up to 18 in the actual radar) coherent pulses with a PRI of $T=K\tau$ seconds where τ is the Nyquist sampling period (i.e. the range cell duration). The STAP is a two-dimensional filter in the "direction of arrival (DOA)-Doppler frequency (f_D)" plane. As a result STAP focuses a main beam towards the target and nulls out the regions of the "DOA- f_D" plane containing the interference.

QRD constitutes the fundamental component of voltage-domain algorithm. It operates recursively by using each snapshot of data to update the on-line estimation of the disturbing environment without forming the interference covariance matrix and only requires $O(N^2M^2)$ arithmetic operations to be performed every sample time. The scheme of Figure 1has been applied to the data recorded by the NRL radar.

7.2 DATA FILES USED IN THE DATA REDUCTION EXPERIMENTS

This section describes the data files, recorded by NRL radar, used for space-time processing experiments. The files refer to ground clutter, land-sea clutter interface, and jamming. The following information have been extracted by the data files, namely: (i) echo power in a radar receiving channel versus range, (ii) the

probability density function (pdf) of the amplitude and phase of the radar echoes, (iii) the eigenvalue spectrum, and (iv) the two-dimensional power spectral density of the clutter versus f_D and DOA. In this paper, just a subset of these information is enclosed.

7.2.1 Ground clutter

Two data files were examined, namely DL050 and DL087. For these files we have calculated the amplitude and phase histograms of the radar echoes. The histograms have been estimated using 896 echoes along range. The amplitude histograms show visually a good fit with the Rayleigh pfd. One more test to verify whether the histogram adequately matches the Rayleigh pdf is to calculate the mean to median ratio. The estimated value is 1.115, while the exact value results 1.442. The histogram of phase is approximately uniform. For file DL050 the spectrum of eigenvalues of the interference covariance matrix is reported in Figure 4. The number of antenna is 8, while the number of PRIs is the parameter of the curves ranging from 1 to 18. The covariance matrix has been estimated by averaging 896 independent samples along range. The maximum eigenvalue has been normalised to 0 dB. The minimum eigenvalue, corresponding to the curve labelled with "18" gives a good estimate of the noise floor in each receiving channel; before normalisation this value is about 10 dB. The clutter plus noise power value amounts to 45 dB in each receiving channel; this value has been determined by averaging along range the received signal on the 1st antenna. Thus the clutter-to-noise power ratio is 35 dB.

7.2.2 Land-sea clutter

Figure 5 portrays the power vs. range of the echoes collected by the 1st antenna for the data DR075. At the 480th range cell the transition from sea to land is clearly visible. The sea clutter power, estimated along the first 200 range cells, amount to 12.8 dB. The land clutter power estimated from 600th to 800th range cells measures 30.2 dB.

7.2.3 Jamming

The data file DW015 refers to jamming overland. The jammer appears at the end fire, i.e. DOA=90°. Figure 6 reports the eigenvalues (normalised to 0 dB) of the estimated covariance matrix (over 300 range cells) for N=8 antenna and M=1 PRI (curve a) and N=8 and M=2 (curve b). The presence of one principal eigenvalue in the curve (a) indicates the presence of one jamming source. We also estimates (over 300 range cells) from the data file that the jammer plus noise power is equal to 36.5 dB. The thermal noise, evaluated by the minimum eigenvalue of the interference covariance matrix is 30 dB. Thus the jammer-to-noise power ratio is 6.5 dB.

7.3 PERFORMANCE EVALUATION

The detection performance of the systolic array of Figure 1 depends upon the array parameters, the interference environment and the target signal features. The parameters defining the trapezoidal array are: (i) the dimension N,M of the data snapshot vector which equal the number of input lines to the triangular systolic canceller, (ii) the forgetting factor of the QRD canceller and (iii) the number L of linear columns for constraints. Synthetic targets as well as signals injected in the receiver are used to determine the integration of target echoes. Performance during steady state are measured in terms of: (a) Improvement Factor (IF), (b) visibility curve, i.e. IF vs. target f_D sweeping across the PRF and (c) the two-dimensional response of the adaptive system versus DOA and f_D.

7.3.1 Performance against ground clutter

Consider the file DL087. Assume to have a trapezoidal array with one antenna (N=1), eighteen pulses (M=18) and L=3 linear columns (processing cells DE of Figure 1). The constraints in the three columns are set to detect a target having the following Doppler frequencies: 0.5 PRF, 0.25 PRF and 0 PRF. A synthetic target

having a Doppler frequency value of 0.5 PRF was added at the 264th range cell. Figure 7a, b and c show the power in dB of the residue signals at the output of the three columns. Note that the target echo appears only in the Figure 7a as expected; the estimated IF is 35 dB.

7.3.2 Performance against sea-land clutter

The file DR075 contains a test target, injected in the receiver at the 3547th range cell. The Doppler frequency of the target is 0.5 PRF and the DOA is 0°. Figure 8 portrays the power in dB of the residue signal obtained by adaptively processing the echoes received by N=8 and M=18 PRIs. The trapezoidal systolic array has one vertical column (L=1) with the constraints f_D=0.5 PRF and DOA=0° which are fully matched to the target signal. The spike appears at the 3691st cell which differs from the original target range due to the space-time filter delay which is equal to the total number of degrees of freedom, i.e. 144.

The visibility curve for a fictitious target having DOA=0° and Doppler frequency sweeping across the radar PRF is reported in Figure 9; the visibility curve is approximately flat except around f_D=0 which is the mean Doppler frequency of clutter after compensation of the platform speed. From visibility curve the maximum IF value amounts to 44 dB, while the optimum IF would be 45.5 dB which is just few dBs higher than the values shown in visibility curve.

7.3.3 Performance against jammer

The improvement factor of an array of N=8 antennas, one PRI (M=1) and one column constraint is shown in Figure 10 as a function of the DOA of a simulated target scanning the angular interval [-90°, +90°]. The jammer is that described in section 7.2.3. It is noted that the maximum IF is about 13 dB, while the optimum IF value would be 17 dB. The 4 dB loss is due to the adaptation of the systolic arrays.

7.4 DETECTION OF VEHICULAR TRAFFIC

The detection of vehicular traffic has been attempted along US route 50 (see, for details, [7]). Four points on the route have been selected (bearing angle relative to the array normal, with positive values coming from the right hand side of the array):

1st point: range=39268 m, azimuth=-5.8°; 2nd point: range=39268 m, azimuth=-3.4°;
3rd point: range=39429 m, azimuth=-0.6°; 4th point: range=39429 m, azimuth=1.0°;

The systolic array processes the snapshots along the range cells received by 8 antennas and 18 PRIs (i.e. it works with the maximum number of adaptive degrees of freedom). The adapted residue along the range cells has been further processed by a CFAR thresholding device based on cell average (CA) technique. The CFAR-CA has two guard range cells on each side of the range cell under test and twenty range cells on each side to estimate the detection threshold. The CFAR-CA has been set to guarantee a P_{FA} of 10^{-4}. Figure 11 depicts the adapted residue vs. range when the receiving antenna pattern is focused at −5.8°, which is the azimuth value corresponding to the 1st point on the US route 50. The analysed Doppler frequency is 0.225 PRF which corresponds to a radial speed of 23.2 m/s (i.e. 83.52 km/h) compatible with vehicular traffic. A detection appears at the 932th range cell that comfortably compares with the expected location of the target. Similar results have been obtained for the other three points on the US route 50 [7].

8. CONCLUDING REMARKS

The research work described in this paper and the enclosed references are also relevant for other radar applications, sometimes simpler than the STAP, namely (i) ground based or ship-borne radars for clutter cancellation and (ii) ground based or ship-borne radars equipped with a multi-channel phased array antenna for

jamming cancellation. The STAP reverts to the first application by setting N=1, while becomes the second application for M=1. Thus, the adaptive processing architectures described in this paper are applicable also to the (i) and (ii) systems. In general, the number of degrees of freedom involved is one order of magnitude less than the STAP case; this makes less critical the implementation of a VLSI based systolic array. A practical application of systolic processing for classical ground based or ship borne radar is described in [17] where it is shown how to combine in one systolic scheme the two functions of adaptive interference cancellation and side lobe blanking. The application of STAP to Synthetic Aperture Radar for detecting and imaging of slowly moving targets is discussed in [5] and [13]. In this respect the procedure to form the SAR image by one-bit processing plays a role; this procedure is also applied in the along track interferometry (ATI) – SAR to detect moving targets [18]. It can be shown that this approach offers a considerable computational advantage; FPGA technology has been successfully applied to implement the one-bit SAR processing. The enormous progress done in the technology for signal processing is under our eyes. Today the key words are: heterogeneous processing (i.e.: based on VLSI, ASIC, FPGA, RISC, MEMS, photonic etc.), virtual and rapid prototyping, modularity and flexibility of processing architectures, re-use and porting of the same, COTS approach to software and hardware, software language (e.g.: System C; Handel C for FPGA), complex design tools like Ptolemy. All these techniques and technologies are conceived to contrast the obsolescence which is one of the most important problems to face today.

9. REFERENCES

[1] A.Farina, "Antenna Based Signal Processing Techniques for Radar Systems", Artech House , 1992.

[2] A. Farina, L. Timmoneri, "Space-time processing for AEW radar", Proc. of Int. Radar Conference, Radar 92, Brighton (UK), 12-13 October 1992, pp. 312-315.

[3] L. Timmoneri, I.K. Proudler, A. Farina, J.C. McWhirter, "QRD-based MVDR algorithm for multipulse antenna array signal processing", IEE Proc. Radar, Sonar and Navigation, Vol. 141, No. 2, April 1994, pp. 93-102.

[4] A. Farina, S. Barbarossa, M. Ceccarelli, A. Petrosino, L. Timmoneri, F. Vinelli, "Application of the extreme eigenvalue analysis to signal and image processing for radar", Invited paper, Colloque International sur le Radar, Paris, 3-6 May 1994, pp. 207-213.

[5] A. Farina, S. Barbarossa, "Space-time-frequency processing of synthetic aperture radar signals", IEEE Trans. on Aerospace and Electronic Systems, vol.30, no. 2, 1994, pp.341-358.

[6] A. D'Acierno, M. Ceccarelli, A. Farina, A. Petrosino, L. Timmoneri, "Mapping QR decomposition on parallel computers: a study case for radar applications", IEICE Trans. on Communications, Vol. E77-B, No. 10, October 1994, pp. 1264-1271.

[7] A. Farina, R. Graziano, F. Lee, L. Timmoneri, "Adaptive space-time processing with systolic algorithm: experimental results using recorded live data", Proc. of the Intl. Conference on Radar, Radar 95, Washington DC, May 8-11, 1995, pp. 595-602.

[8] A. Farina, L. Timmoneri, "Antenna based signal processing techniques and space-time processing". Tutorial, Int. Conference, Radar 95, Washington D.C. (USA), May 8-11, 1995.

[9] P. Kapteijin, E. Deprettere, L. Timmoneri, A. Farina, "Implementation of the recursive QR algorithm on a 2x2 CORDIC test-board: a case study for radar application", Proc. of the 25th European Microwave Conference, Bologna (Italy), 1995, pp. 490-495.

[10] A. Farina, A. Saverione, L. Timmoneri, "The MVDR vectorial lattice applied to space- time processing for AEW radar with large instantaneous bandwitdh", IEE Proc. Radar, Sonar and Navigation, Vol. 143, No. 1, February 1996, pp. 41-46.

[11] A. Farina, L. Timmoneri, "Parallel algorithms and processing architectures for space-time adaptive processing", CIE - International Conference on Radar - ICR96, Beijing (P. R. of China), Invited Paper for the Workshop on STAP, 1996, pp. 771-774.

[12] P. Bollini, L. Chisci, A. Farina, M. Giannelli, L. Timmoneri, G. Zappa, "QR versus IQR algorithms for adaptive signal processing: performance evaluation for radar applications", Proc. of IEE on Radar, Sonar and Navigation, October 1996, Vol. 143, No. 5, pp.328-340.

[13] P. Lombardo, A. Farina, "Dual antenna baseline optimization for SAR detection of moving targets", Proc. of ICSP96, Beijing, P.R. China, pp. 431-433.
[14] A. Farina, P. Lombardo: "Space-time adaptive signal processing", Tutorial, Int. Conference on Radar, Radar 97, Edinburgh (UK), 13 October 1997.
[15] A. Farina, L. Timmoneri, "Real time STAP techniques", Proc. of IEE Symposium on Space Time Adaptive Processing, London, 6th April 1998, pp. 3/1-3/7.
[16] A. Farina, L. Timmoneri, "Real time STAP techniques", Electronic, Communication and Engineering Journal (IEE) Special Issue on STAP, Vol. 11, No.1, February 1999, pp. 13-22.
[17] A. Farina, L. Timmoneri, "Systolic schemes for joint SLB, SLC and adaptive phased-array", Proc. of the IEEE Intl. Conference Radar 2000, Washington DC, 7-12 May 2000, pp. 602-607.
[18] V. Pascazio, G. Schirinzi, A. Farina, "Moving Target Detection by Along Track Interferometry", IGARSS 2001, Sidney (Australia), July 2001.

10. ACKNOWLEDGEMENTS

The Author sincerely acknowledges with thanks the cooperation of His Colleague Dr. L. Timmoneri (AMS) that has greatly contributed to the work described in this paper. Other Colleagues, duly quoted in the References, have noticeably contributed to the research.

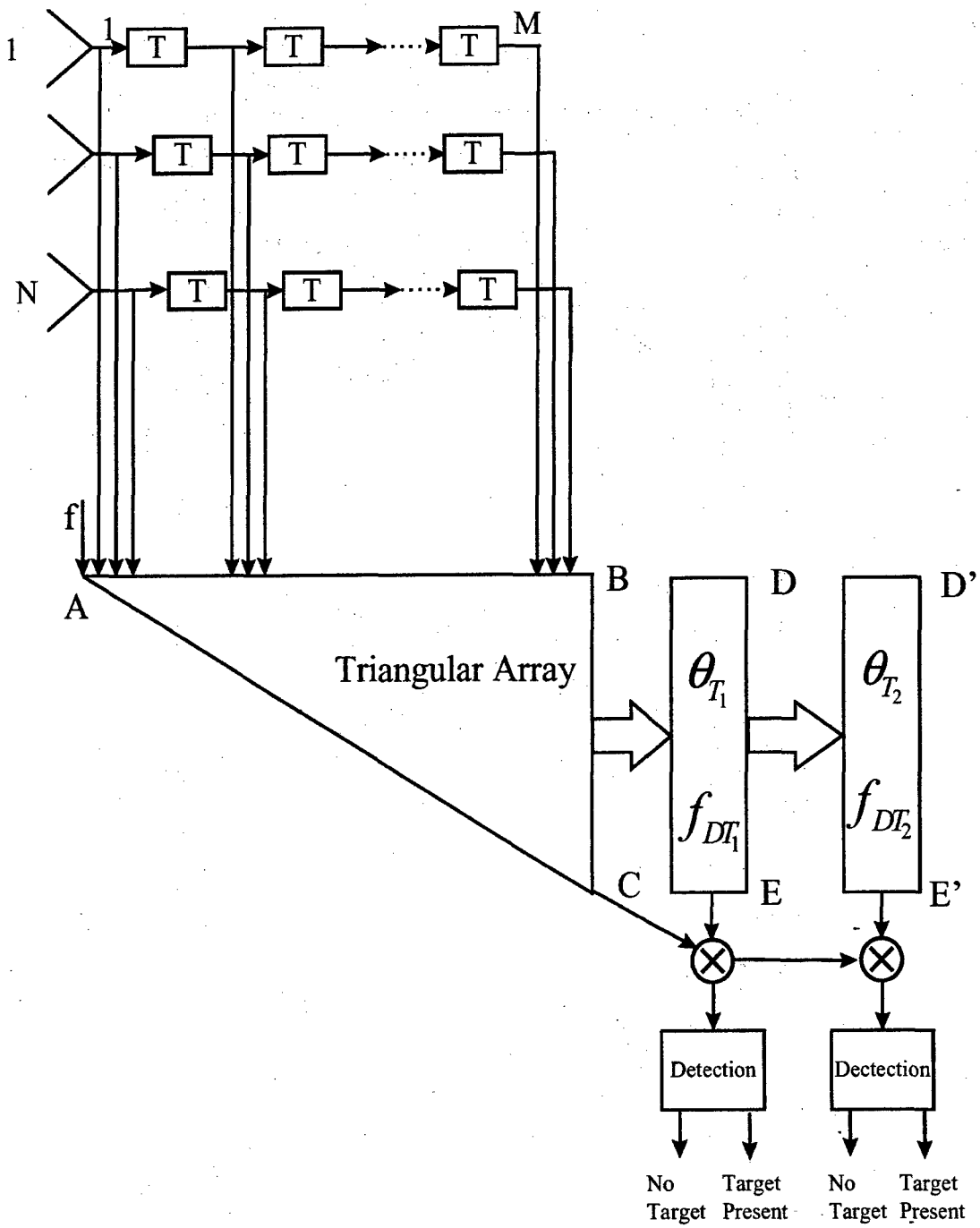

Figure 1: Baseline QRD based MVDR flow graph (after [3]).

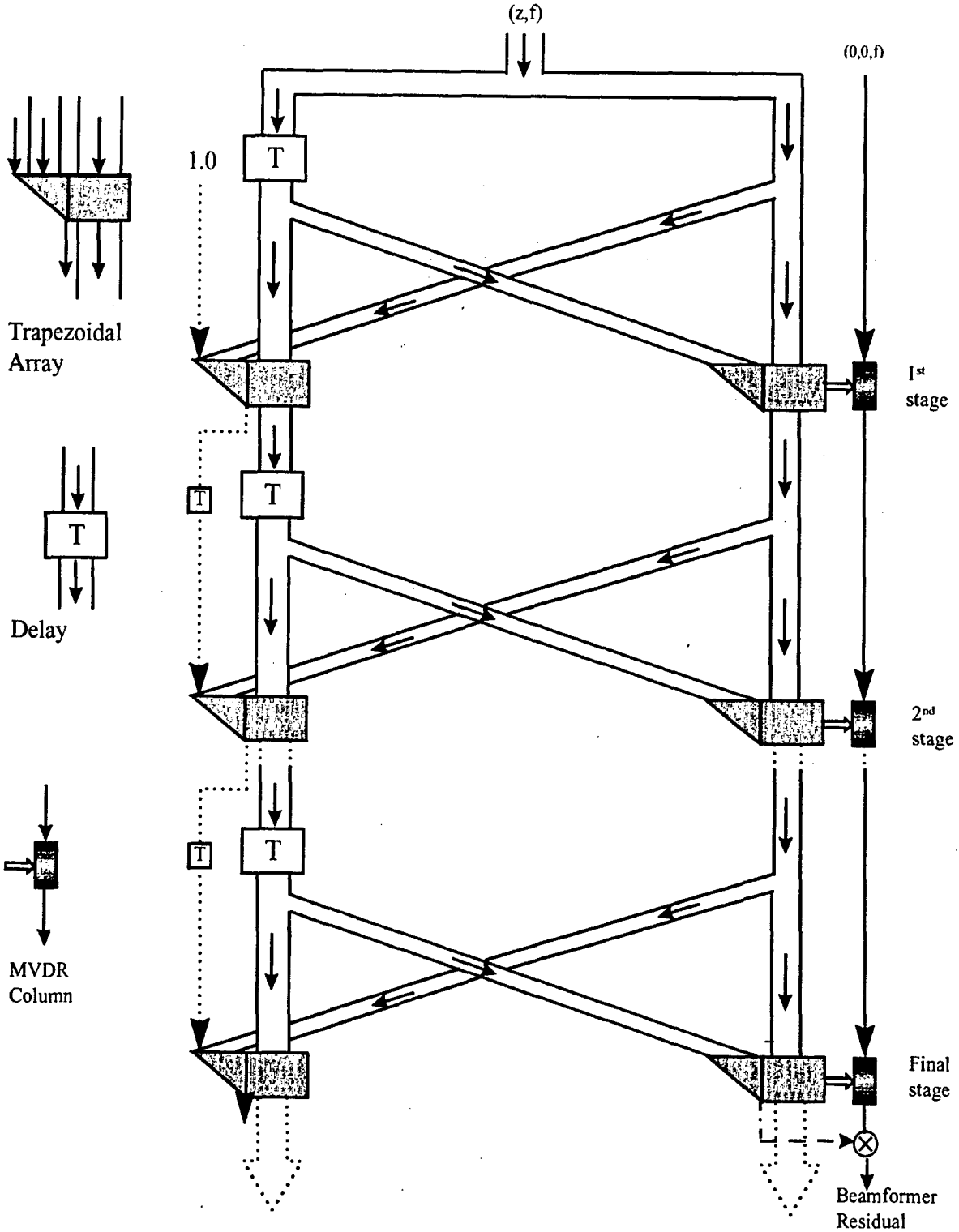

Figure 2: MVDR lattice processor (after [3]).

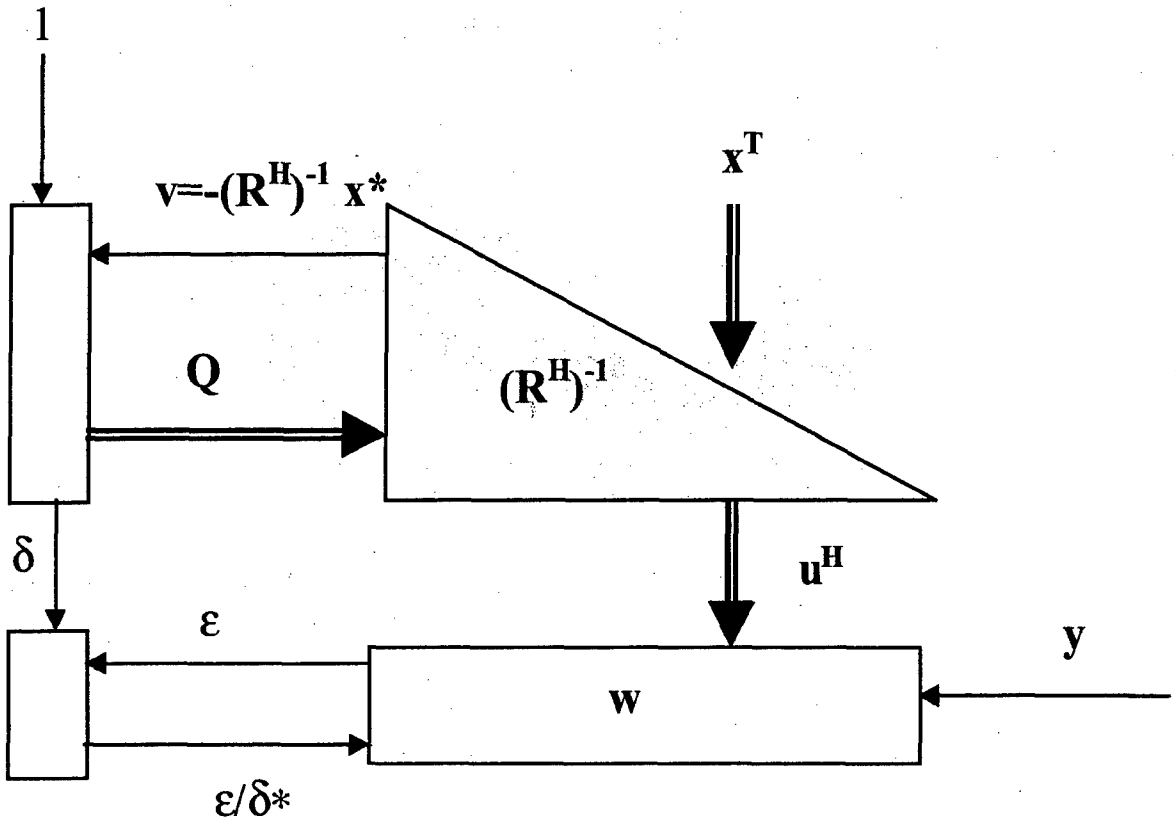

Figure 3: RLS-IQR array (after [12]).

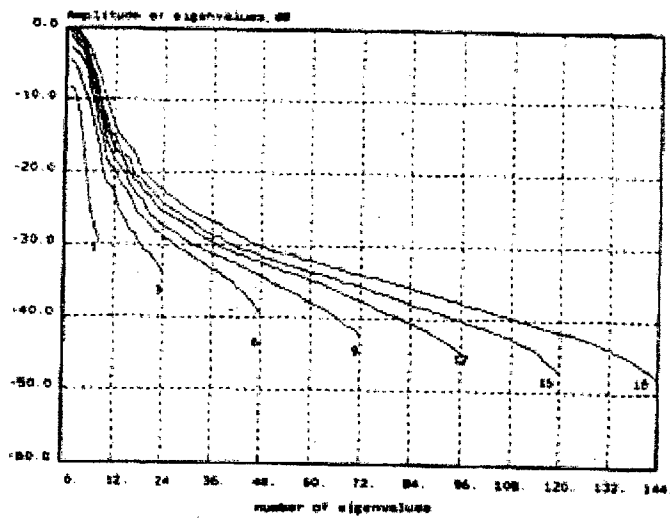

Figure 4: Eigenvalue spectrum for data file DL050 (ground clutter) (after [7]).

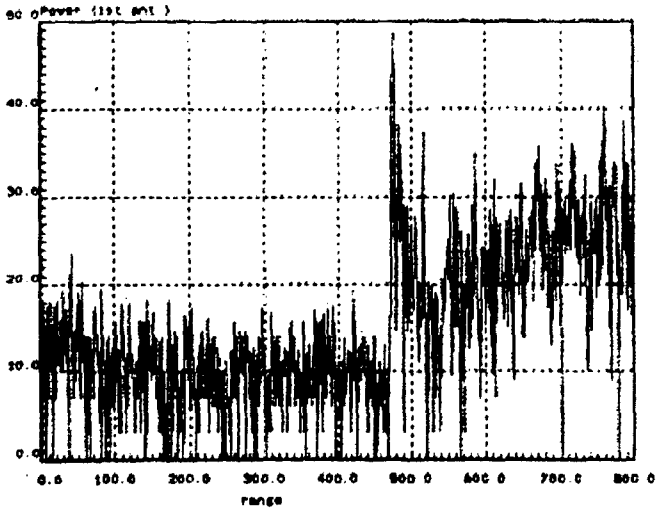

Figure 5: Power versus range of the radar echoes collected by the 1st antenna of the array (after [7]).

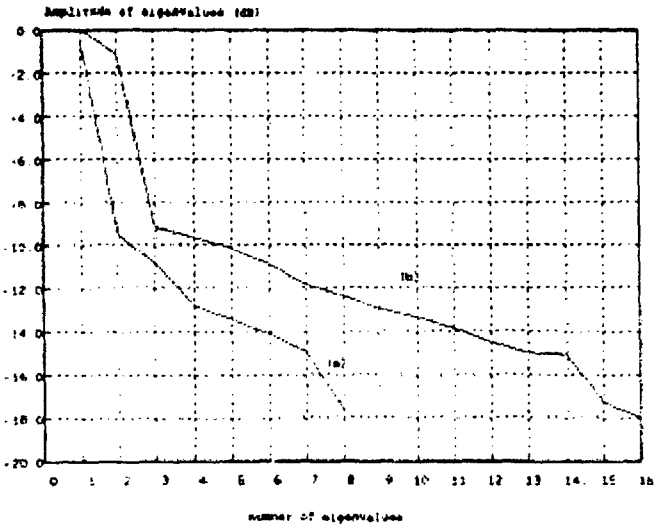

Figure 6: Spectrum of eigenvalues of jamming interference. Curve a: N=8 antennas and M=1 PRI; curve b: N=8, M=2 (after [7]).

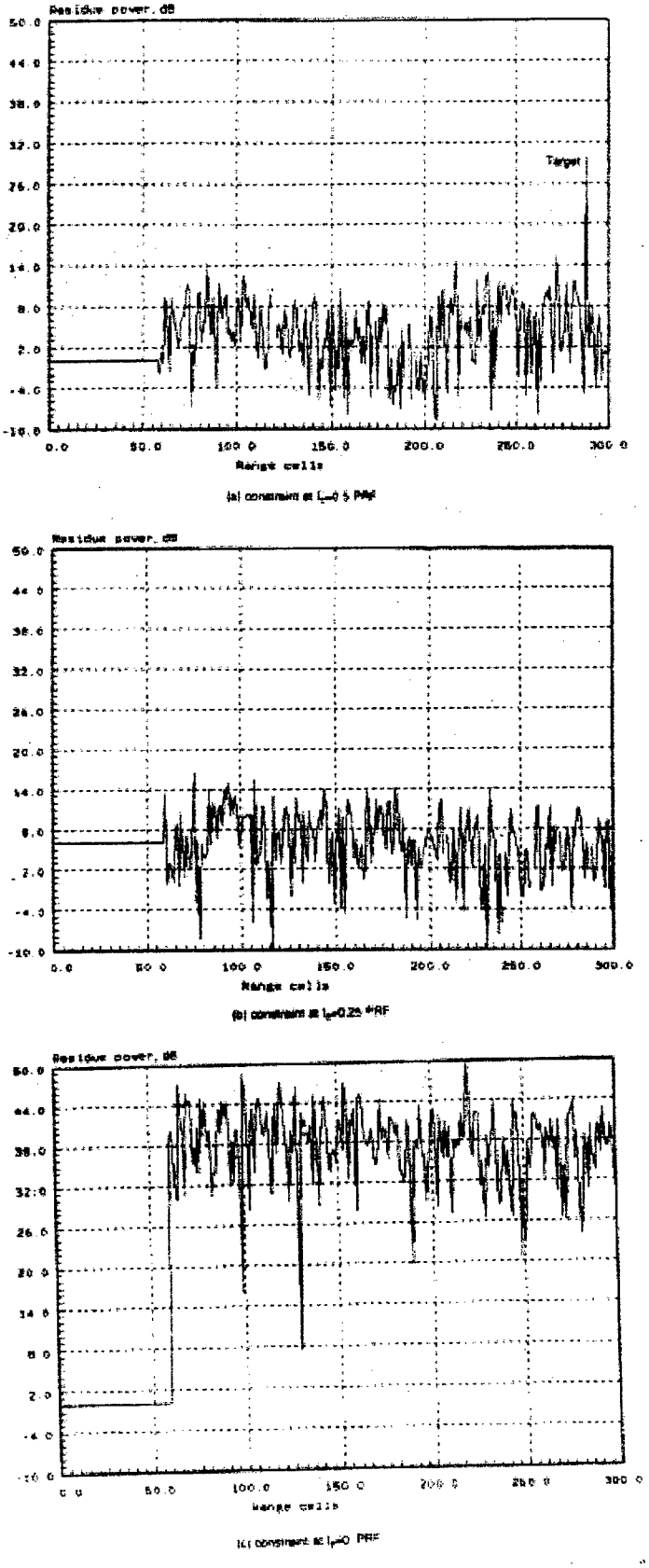

Figure 7: Processing of ground clutter live data (after [7]).

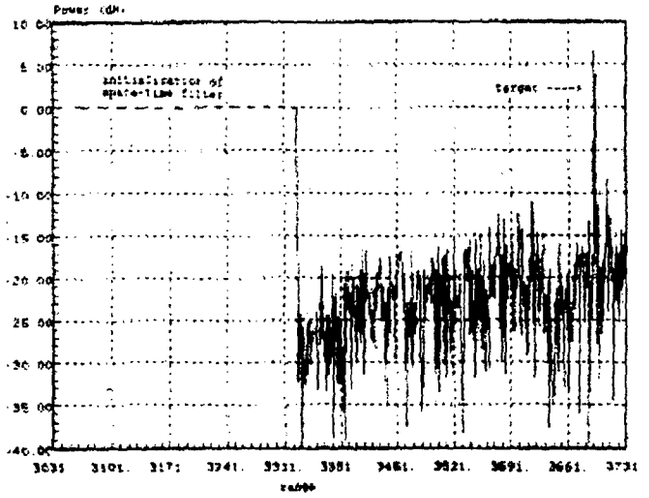

Figure 8: Power of residue signal for data file DR075 (after [7]).

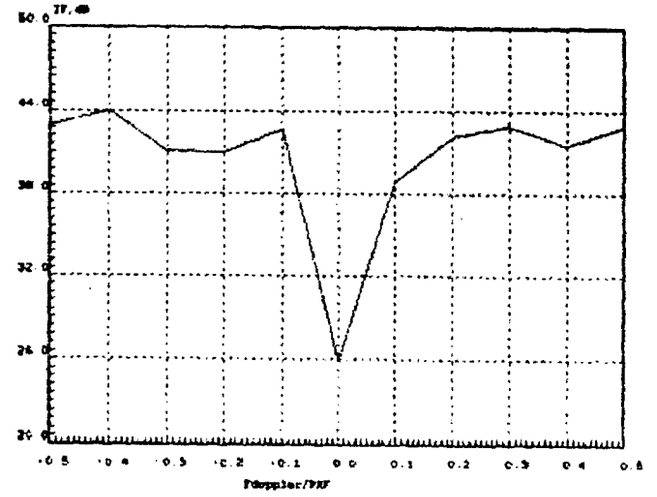

Figure 9: Visibility curve for data file DR075 (after [7]).

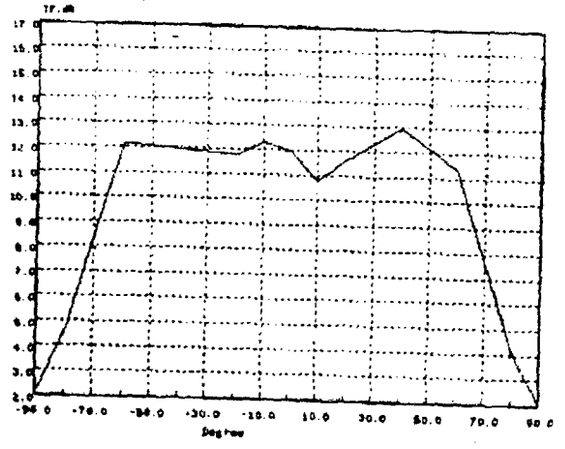

Figure 10: IF vs. DOA of a simulated target (after [7]).

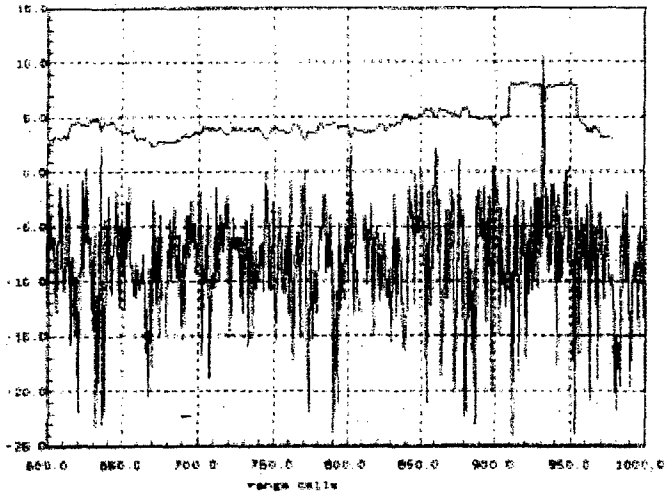

Figure 11: Adapted residue power and detection threshold curves vs. range cell (after [7]).

Pros	Cons
programmable & flexible	complex infrastructure including I/O control and protocols
robust to technology obsolescence	high speed data buses
re-use of previously developed software	high speed memory and memory control
essential in design trajectory of VLSI custom architecture (search for trade-off between flexibility and modularity, parallelisation options)	multi-DSP infrastructure requires extra-overhead which brings to a decline of ideal linear increment of computational power.

Table 1 : Pros and cons of COTS.

Pros	Cons
extremely high throughput (bulk processing)	low degree of flexibility
limited size and power consumption	expensive for limited number of pieces to produce

Table 2 : Pros and cons of VLSI.